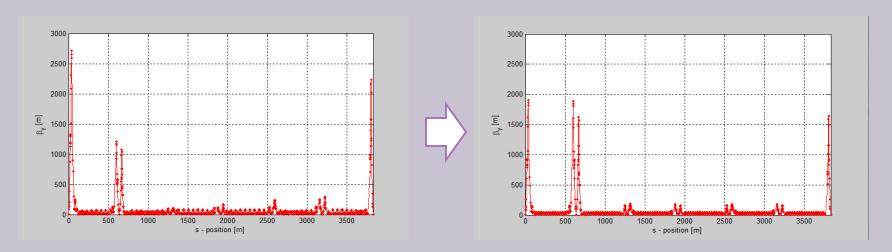
Orbit Response Matrix Commissioning at RHIC

APEX Workshop 2010 October 18, 2010

Tasha Summers BNL-CAD

The Purpose of ORM

- Find and correct errors in storage ring linear optics
- Compare a measured orbit response matrix to the model
 - Fit parameters in model are adjusted until model matches measured
 - Find BPM and corrector offsets
 - Find quadrupole strength errors and rolls
 - Restore lattice symmetry



Generating the Data

- Start with list of correctors calculate size of kick to get maximum response in arcs
- Measure baseline orbit, change corrector in each ring, measure new orbit
- Measure Horizontal and Vertical planes separate, but Blue and Yellow rings together
- Takes 2 hours to complete both planes
- Need to save machine lattice files (design)

- 10 ./orm.pl bi1-th17 yo1-th16 0.0203516200625966 0.0205356343550386 11 ./orm.pl bi1-th19 yo1-th18 0.0206769848740387 0.0204098634496941
- 12 ./orm.pl bi1-th21 yo1-th20 0.020355751729253 0.020563924729105
 13 ./orm.pl bo2-th20 yi2-th19 0.0200861129826991 0.020565359714291
- 14 ./orm.pl bo2-th18 yi2-th17 0.0208603996964928 0.0204966451915098
- 14 ./orm.pl bo2-th18 yi2-th17 0.0208603996964928 0.0204966451915098 15 ./orm.pl bo2-th16 yi2-th15 0.0201407520140505 0.0205249906768779
- 16 ./orm.nl ho2-th14 vi2-th13 0.0207715567914651 0.0205376697879686



1 bi1-th3 -0.000101461143544 2 vo1-th2 8.42926695838e-05 3 Blue bi1-th3 0 baseline Wed Jan 6 20:03:43 EST 2010 4 Wed_Jan__6_20:03:17_2010.sdds 5 Wed_Jan__6_20:03:34_2010.sdds 6 Wed_Jan__6_20:03:36_2010.sdds 7 Wed_Jan__6_20:03:37_2010.sdds 8 Yellow yo1-th2 0 baseline Wed Jan 6 20:03:43 EST 2010 9 Wed_Jan__6_20:03:18_2010.sdds 10 Wed_Jan__6_20:03:34_2010.sdds 11 Wed_Jan__6_20:03:36_2010.sdds 12 Wed_Jan__6_20:03:37_2010.sdds 13 setting bi1-th3 to -9.37172403268711e-05, yo1-th2 to 9.2911309865437e-05 (real) 14 Blue bi1-th3 7.74390321712892e-06 Wed Jan 6 20:03:56 EST 2010 15 Wed_Jan__6_20:03:37_2010.sdds 16 Wed_Jan__6_20:03:48_2010.sdds 17 Wed_Jan__6_20:03:49_2010.sdds 18 Wed_Jan__6_20:03:51_2010.sdds 19 Yellow yo1-th2 8.61864028163694e-06 Wed Jan 6 20:03:56 EST 2010 20 Wed_Jan__6_20:03:37_2010.sdds 21 Wed_Jan__6_20:03:48_2010.sdds 22 Wed_Jan__6_20:03:49_2010.sdds 23 Wed_Jan__6_20:03:51_2010.sdds 24 setting bi1-th3 to -0.000101461143544, yo1-th2 to 8.42926695838e-05 (real) 25 bi1-th5 3.11075502691e-05 26 yo1-th4 2.89973452571e-05 27 Blue bi1-th5 0 baseline Wed Jan 6 20:04:11 EST 2010

28 Wed_Jan__6_20:03:51_2010.sdds 29 Wed_Jan__6_20:04:02_2010.sdds

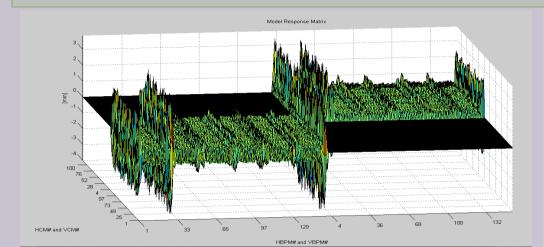
Assembling the Data

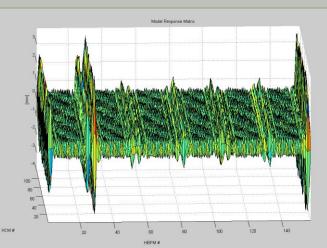
```
Blue-Au102-store-design.sxf
// SXF version 1.0
ring sequence
   "g6-markx"
      tag = g6-markx
      at = 0.00000000000e+00
  "gố-bx"
    monitor (
      at = 7.99313400000e+00
  eldxmp06
    multipole {
      tag = eldxmp06
at = 9.80000000000e+00
  "gó-dhx"
    sbend {
      tag = g6-dhx
at = 1.16501096878e+01
      1 = 3.70021937559e+00
        k1 = Γ -1.88579569387e-02 1
        k1 = [ 0.0 0.00000000000e+00 5.16443668467e-02 ]
        k1 = [ 0.0 -0.000000000000e+00 -7.33998067962e-04 ]
      exit.dev
         k1 = [ 0.0 -0.00000000000e+00 -1.66735363586e-03 ]
```



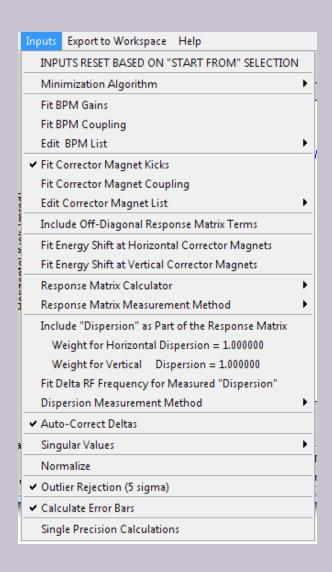
```
0.471 1.635 1.432 -0.132 -0.566 -0.227 -0.130 0.664 0.024 -0.701 0.055 0.681 -0.157 -0.664 0.231 0.626 -0.323 -0.587 0.380 0.535 0.026 -0.550 -0.486 2.764
                                                                                             .509 0.138 -0.529 -0.061 0.529 -0.060 -0.163 -0.444 -0.123 1.19
     % Element definitions:
                                                                                             532 -0.655 -0.452 0.719 0.355 -0.859 -0.313 0.021 0.472 -1.876
-0.4
                                                                                             .192 -0.710 0.270 0.658 -0.361 -0.635 -0.058 0.561 0.539 -3.001
                                                                                             -0.552 0.640 0.449 -0.728 -0.379 0.855 0.316 -0.008 -0.469 1.84
     g6markx = marker('g6markx', 'IdentityPass');
0.5
                                                                                             0.271 0.562 -0.367 -0.529 0.404 0.457 -0.002 -0.544 -0.447 2.59
     DRIFT1 = drift('DRIFT1', 7.993134, 'DriftPass');
                                                                                             14 -0.739 -0.426 0.789 0.309 -0.931 -0.319 0.085 0.532 -2.216
    g6bxh = hmonitor('g6bxh', 'IdentityPass');
-0.1
                                                                                             1.354 -0.483 0.401 0.408 -0.461 -0.346 0.041 0.536 0.375 -2.310
-0.5
     g6bx = monitor('g6bx', 'IdentityPass');
                                                                                              -0.458 0.793 0.332 -0.865 -0.250 0.970 0.309 -0.164 -0.590 2.5
                                                                                             .394 0.366 -0.461 -0.317 0.475 0.214 -0.081 -0.511 -0.299 1.984
     g6bxv = vmonitor('g6bxv', 'IdentityPass');
                                                                                             .028 0.799 -0.097 -0.803 0.170 0.809 0.152 -0.474 -0.611 3.136
    DRIFT2 = drift('DRIFT2', 1.806866, 'DriftPass');
0.89
                                                                                             511 -0.427 -0.569 0.495 0.487 -0.660 -0.302 -0.180 0.291 -0.857
-0.1
     g6dhx = sbend('g6dhx', 3.70021937559e+00, -1.88579569387e-02, 0, 0, 0, 'BndMPoleSymplect).080 -0.752 0.154 0.714 -0.260 -0.719 -0.107 0.505 0.565 -3.01
                                                                                             -0.610 0.514 0.523 -0.612 -0.465 0.750 0.314 0.109 -0.372 1.282
     erdxmpO6 = quadrupole('ELDX', 1e-10, -961487.901076, 'StrMPoleSymplectic4Pass');
                                                                                             3.135 0.689 -0.247 -0.670 0.303 0.635 0.071 -0.523 -0.535 2.910
     DRIFT3 = drift('DRIFT3', 7.006893798955, 'DriftPass');
                                                                                             580 -0.612 -0.516 0.673 0.414 -0.833 -0.325 -0.044 0.434 -1.621
     bo6dh0 = sbend('bo6dh0', 3.58896623069e+00, 1.51804205016e-02, 0, 0, 0, 'BndMPoleSymplec', 247 -0.591 0.315 0.538 -0.384 -0.494 -0.017 0.529 0.454 -2.61
-0.1
    DRIFT4 = drift('DRIFT4', 0.927703761354999, 'DriftPass');
                                                                                             -0.565 0.624 0.479 -0.700 -0.396 0.841 0.320 0.020 -0.449 1.725
                                                                                             548 0.055 -0.562 0.021 0.547 -0.157 -0.192 -0.425 -0.066 0.931
    bo6b1h = hmonitor('bo6b1h', 'IdentityPass');
                                                                                             1.358 -0.455 0.423 0.397 -0.460 -0.318 0.051 0.532 0.360 -2.259
    bo6b1 = monitor('bo6b1', 'IdentityPass');
                                                                                             .457 0.260 -0.484 -0.186 0.508 0.081 -0.121 -0.476 -0.212 1.581
    bo6b1v = vmonitor('bo6b1v', 'IdentityPass');
                                                                                             J.212 -0.600 0.281 0.551 -0.349 -0.516 -0.030 0.509 0.458 -2.60
    DRIFT5 = drift('DRIFT5', 0.336944800000001, 'DriftPass');
                                                                                              -0.280 0.870 0.160 -0.895 -0.060 0.976 0.258 -0.331 -0.648 3.0
                                                                                             542 0.076 -0.556 -0.002 0.542 -0.133 -0.185 -0.430 -0.081 0.999
     bo6qd1 = quadrupole('bo6qd1', 1.44000000000e+00, -0.0575190064271528, 'StrMPoleSymplecti
0.2
                                                                                             70 -0.885 -0.174 0.897 0.047 -0.988 -0.260 0.339 0.653 -3.089
     DRIFT6 = drift('DRIFT6', 1.182428, 'DriftPass');
                                                                                             -D 586 D 035 D 562 -D 141 -D 555 D 264 D 219 D 38D -D 011 -D 55
    bo6th2 = corrector('bo6th2',0, [0.00000000e+00 0.00000000e+00], 'CorrectorPass');
                                                                                              -0.206 0.861 0.069 -0.890 0.013 0.939 0.227 -0.388 -0.652 3.17
                                                                                             36 -0.170 -0.596 0.242 0.546 -0.403 -0.254 -0.332 0.097 0.140 0
     DRIFT7 = drift('DRIFT7', 0.494203500000001, 'DriftPass');
    bo6qf2 = quadrupole('bo6qf2', 3.39163300000e+00, 0.0560097755243566, 'StrMPoleSymplectic' 100 -0.847 -0.012 0.834 -0.108 -0.884 -0.191 0.436 0.631 -3.15
                                                                                              -0.622 0.304 0.563 -0.408 -0.527 0.545 0.280 0.244 -0.215 0.44
    DRIFT8 = drift('DRIFT8', 1.3709615, 'DriftPass');
                                                                                              -0.454 0.846 0.320 -0.916 -0.228 1.022 0.316 -0.208 -0.634 2.7
0.8: | bo6qs3 = quadrupole('bo6qs3', 1.00000000000e-06, 0, 'StrMPoleSymplectic4Pass');
                                                                                              .502 0.185 -0.538 -0.120 0.532 -0.010 -0.153 -0.474 -0.166 1.39
0.439 1.171 0.970 -0.340 0.234 0.365 0.772 -0.545 -0.679 0.631 0.575 -0.744 -0.496 0.792 0.367 -0.872 -0.273 0.897 0.142 -1.010 -0.293 0.265 0.640 -2.916
-0.874 -2.753 -2.380 0.400 0.486 -0.008 -0.429 -0.384 0.449 0.325 -0.502 -0.258 0.522 0.163 -0.561 -0.098 0.550 -0.006 -0.561 0.121 0.184 0.437 0.087 -1.03
-0.308 -0.755 -0.626 0.276 -0.299 -0.363 -0.714 0.581 0.589 -0.596 -0.515 0.760 0.391 -0.829 -0.303 0.863 0.166 -0.906 -0.080 0.977 0.264 -0.321 -0.649 3.0
```

Machine list and archived files -> Matlab lattice file and text response matrix

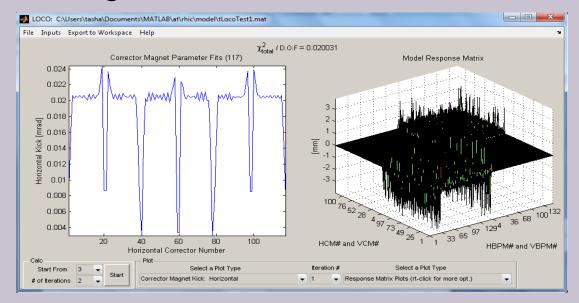




LOCO

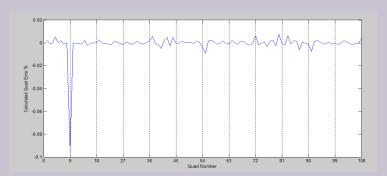


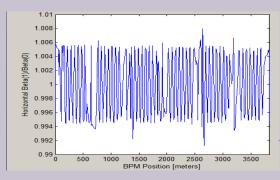
- Uses Matlab Accelerator Toolbox tracking code (developed at SSRL)
- LOCO first written in Fortran, re-written in Matlab to make use of easy matrix calculations and AT
 - used extensively at lightsources
- Gradient search, least squares minimization using SVD to match model to measurement

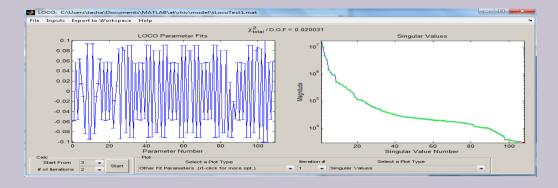


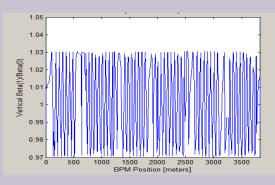
LOCO Simple Test

- Question: can LOCO find known errors in this lattice?
- Added 1 error at bo6-qd9 1%
 3% vertical beta beat, < 1% horizontal
- Fit to quads 1-9 in each sector
- Start: $\chi^2/\text{dof} = 0.1$ First iteration $\chi^2/\text{dof} = 0.02$









LOCO Simple Test 2

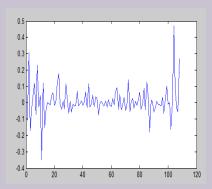
Question: what about an error in the low-β* triplet?

Added 1 error at bo6-qf2 – 0.5%

- Fit to quads 1-9 in each sector
- Start:

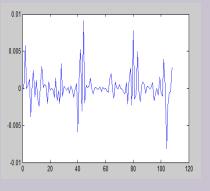
$$\chi^2/dof = 798.9$$

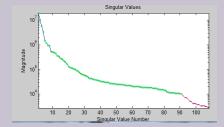
First iteration
 $\chi^2/dof = 3056$
Second iteration
 $\chi^2/dof = 5189$



Added 1 error at bo6-qf2 - 0.1%

- 10% vertical beta beat,20% horizontal
- Start: $\chi^2/\text{dof} = 5.1$ First iteration $\chi^2/\text{dof} = 0.168$ Second iteration $\chi^2/\text{dof} = 0.022$



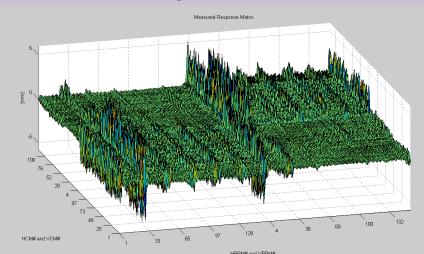


LOCO cannot resolve errors between triplet quads

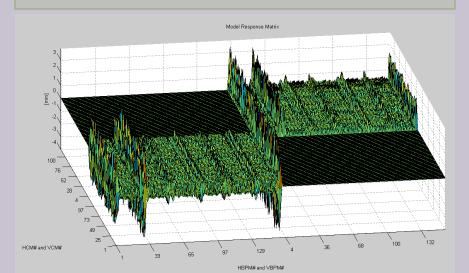
– phase advance is too small

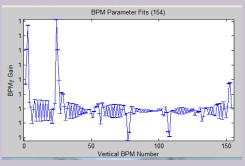
For a good fit, need to have both HBPM and VBPM between each quad

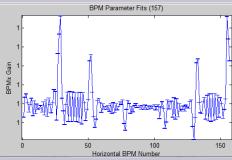
January 2010 Au102 Store, Blue Ring

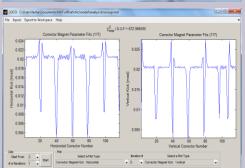


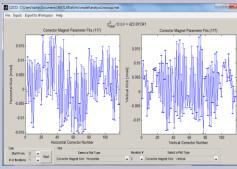
Measured ORM and Model

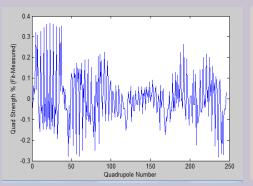


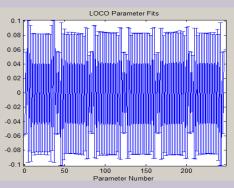












Selecting all options and pressing 'Start' doesn't really work...

Next:

- Lots more simulation
 - Find which quads are best to fit and where degeneracies are
 - Random gradient errors in quads as BPM noise changes –
 determine how much orbit averaging is needed
 - Have Loco solve for random errors in BPMs and corrector kicks
 - Try finding errors in the arc quads
 - Start looking at the coupled matrix
- Should take data with every major lattice change
- Compare responses at store before and after beta squeeze
- See T. Satogata and J. Niedziela C-A/AP Note # 267 February 2007
 Simulations of RHIC Orbit Response Analysis using LOCO